

Container Development

Focus Area: Front-end



Group Workshop

Back-end

Services
IntegrationContent
ServicesAdvanced
Security

Targeting

Publishing

Front-end

Widget
Development

Widget Extras

Container
Development

Foundation

Portal Essentials

Portal Technologies

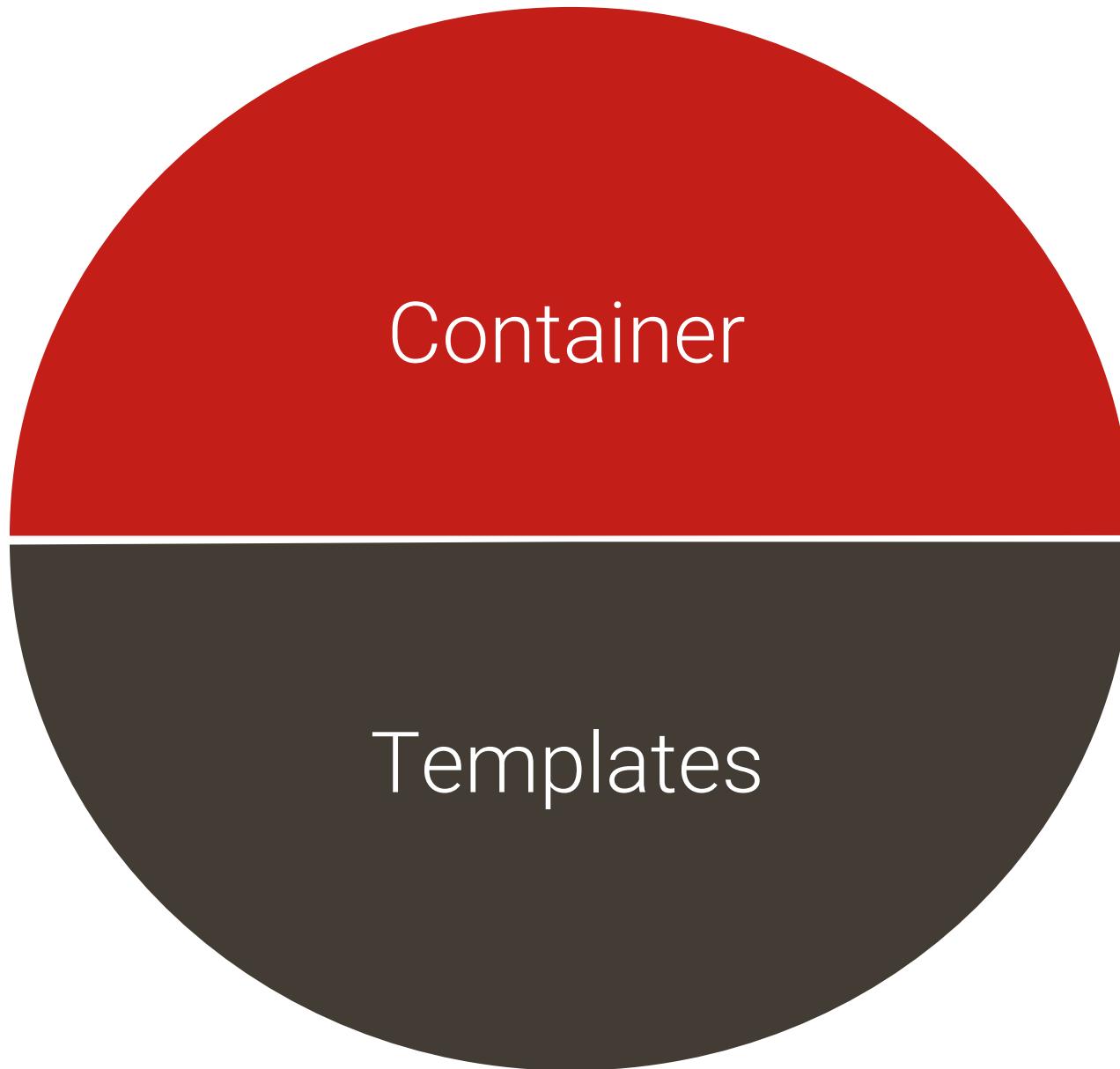
Portal Tools

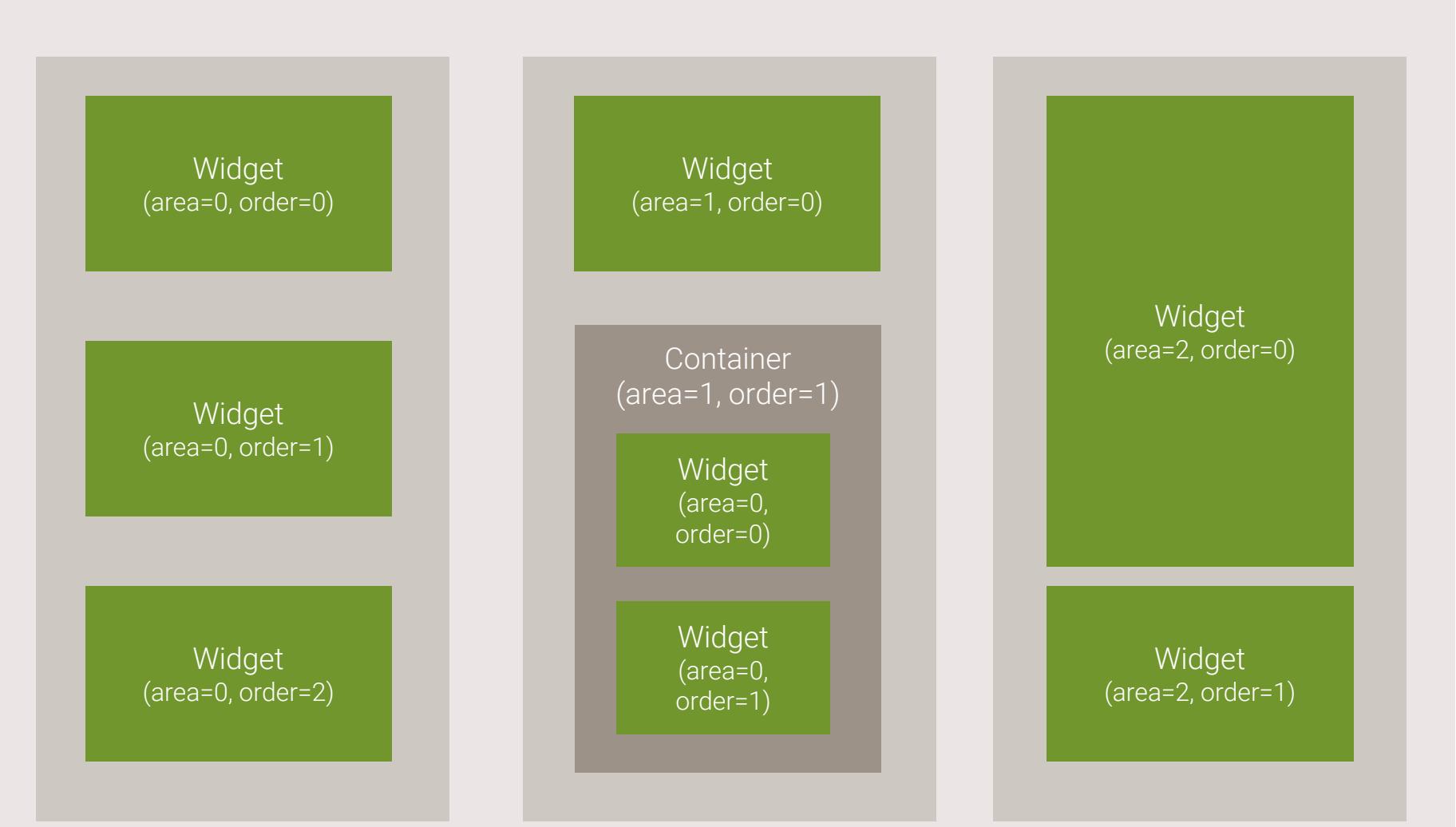
Portal APIs

1. Understanding Containers
2. Learn how to Develop a Container

Understanding Containers

Container Development





Widget
(area=0, order=0)

Widget
(area=0, order=1)

Widget
(area=0, order=2)

Widget
(area=1, order=0)

Container
(area=1, order=1)

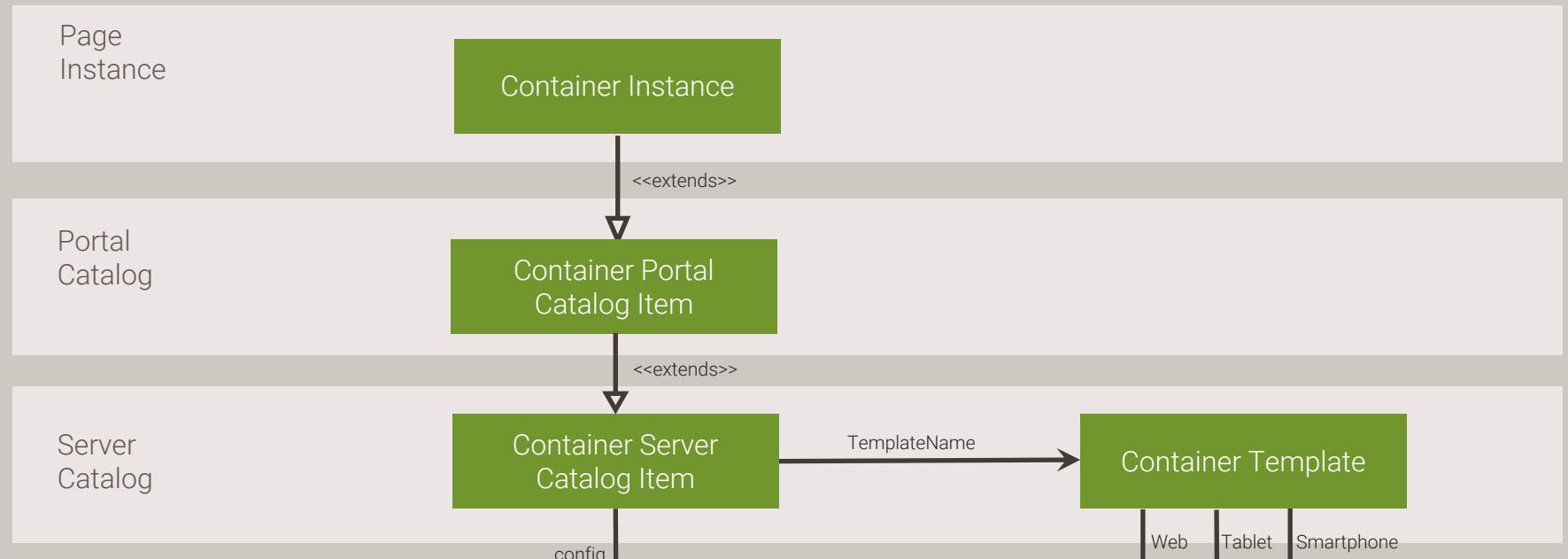
Widget
(area=0,
order=0)

Widget
(area=0,
order=1)

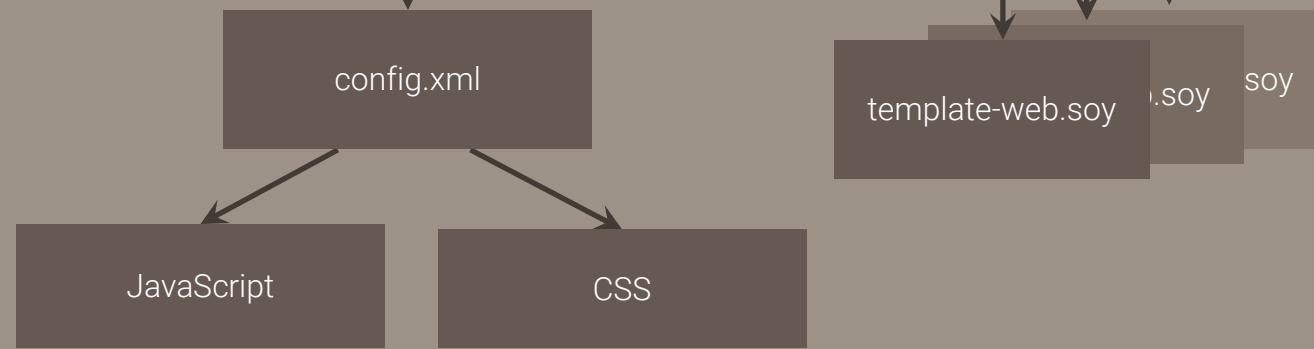
Widget
(area=2, order=0)

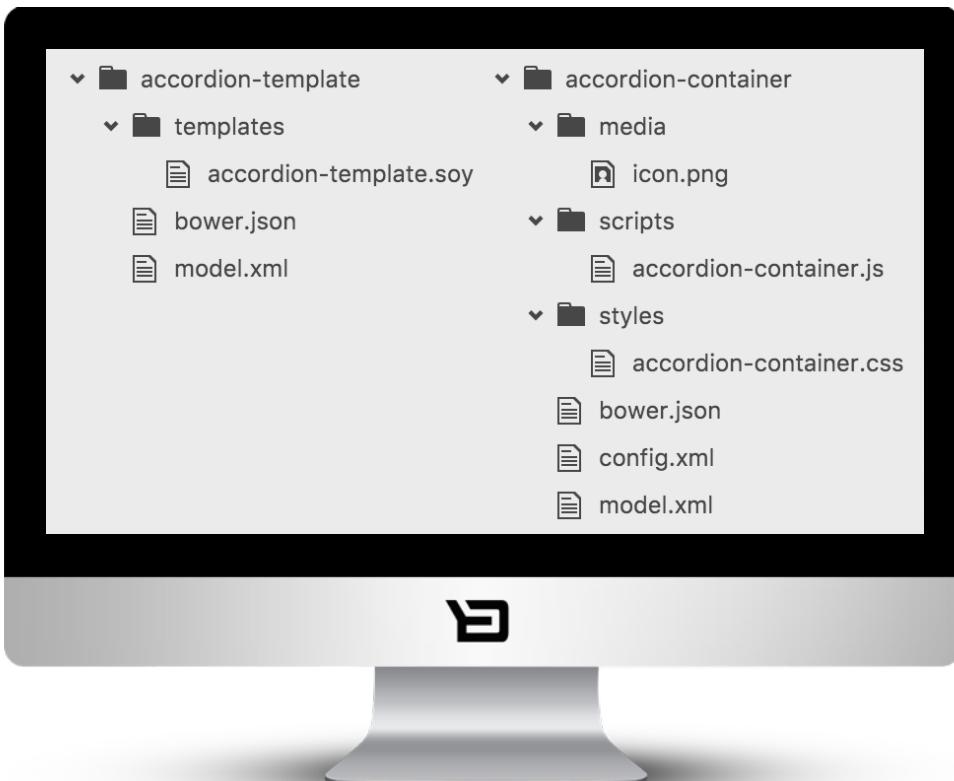
Widget
(area=2, order=1)

Model



Resources





- Google Closure (soy) as content
- JS for functionality
- CSS for styling
- XML for configuration and definition of resources

Scaffold a Container



To learn more about Container Development, we will develop an **Accordion Container**. Let start by scaffolding a Container.

1. Generate a Container and the Template in the **prebuilt** folder.
Note: Generating a Container will ask for the **Template Name**. This needs to match the scaffolded Template

1. Package and Import to Test it on a Page by putting a Widget into the Container.

Container Definition

Container Development



Soy Template



config.xml

```
{namespace templates_templateContainerTrainingAccordion}

/**
* Accordion Container
* @param item Item in context
*/
{template .templateContainerTrainingAccordion}
<div data-pid="${item.name}" class="bp-container bp-ui-dragRoot">
    <div class="accordion--area bp-area">
        {foreach $child in $item.children}
            {if parseInt($child.preferences.area.value) == 0}
                {include($child) | noAutoescape}
            {/if}
        {/foreach}
    </div>
</div>
{/template}
```

```
<?xml version="1.0" encoding="UTF-8"?>
<w3c:widget xmlns:w3c="http://www.w3.org/TR/widgets"
  xmlns:bb="http://www.backbase.com/ns/widgets"
  xmlns="http://www.w3.org/1999/xhtml">

  <bb:resources>
    ...
  </bb:resources>
</w3c:widget>
```

Container Functionality

Container Development

```
(function () {
    var Container = b$.bdom
        .getNamespace('http://backbase.com/2013/portalView')
        .getClass('container');

    Container.extend(function (bdomDocument, node) {
        Container.apply(this, arguments);
        this.isPossibleDragTarget = true;
    }, {
        localName: 'templateContainerTrainingAccordion',
        // the same as the viewNamespace property in the import XML for the container
        namespaceURI: 'templates_templateContainerTrainingAccordion'
    }, {
        handlers: {
            DOMReady: function () {
                // triggers when the container is rendered
            }
        }
    });
})();
```

```
<?xml version="1.0" encoding="UTF-8"?>
<w3c:widget xmlns:w3c="http://www.w3.org/TR/widgets"
  xmlns:bb="http://www.backbase.com/ns/widgets"
  xmlns="http://www.w3.org/1999/xhtml">

  <bb:resources>
    <script type="text/javascript" src="js/accordion.js"></script>
  </bb:resources>
</w3c:widget>
```

Implement the Accordion UI



To learn more about Container Development, we will develop an **Accordion Container**. Develop the accordion functionality.

1. Increase your areas to 3 (The first one should be open and the others closed by default, use Bootstrap <http://getbootstrap.com/javascript/#collapse>)

1. Make sure there is a title for each area (configurable via a preference).

2. When clicking the title, the specific area should become visible and the others should be hidden.

Client-Side Rendering

Container Development

```
<?xml version="1.0" encoding="UTF-8"?>
<w3c:widget xmlns:w3c="http://www.w3.org/TR/widgets"
  xmlns:bb="http://www.backbase.com/ns/widgets"
  xmlns="http://www.w3.org/1999/xhtml">

  <bb:resources>
    <link type="text/css" rel="stylesheet" href="css/accordion.css"/>
    <script type="text/javascript" src="js/accordion.js"></script>
    <script type="text/javascript"
      src="$(contextRoot)/templates/AccordionTemplate/template.js"></script>
  </bb:resources>
</w3c:widget>
```

```
(function () {
    var Container = b$.bdom
        .getNamespace('http://backbase.com/2013/portalView')
        .getClass('container');

    Container.extend(function (bdomDocument, node) {
        Container.apply(this, arguments);
        this.isPossibleDragTarget = true;
    }, {
        localName: 'templateContainerTrainingAccordion',
        // the same as the viewNamespace property in the import XML for the container
        namespaceURI: 'templates_templateContainerTrainingAccordion'
    }, {
        template: function(json) {
            var data = {item: json.model.originalItem};
            return window[this.namespaceURI][this.localName](data);
        },
        handlers: {
            DOMReady: function () {
                // triggers when the container is rendered
            }
        }
    });
})();
```

Enabling Client-side Rendering

```
<catalog>
  <container>
    <name>container-training-accordion</name>
    <contextItemName>[BBHOST]</contextItemName>
    <properties>
      ...
      <property name="csr">
        <value type="boolean">true</value>
      </property>
      ...
    </properties>
  </container>
</catalog>
```

Give us your feedback for this training!

Please find the link in the calendar invitation for this training

If not there please use:

<https://is.gd/z8ha74>



Thank you!

Thank you!

academy@backbase.com

www.backbase.com